METHOD AND SYSTEM FOR USING DYNAMIC RANDOM ACCESS MEMORY AS CACHE MEMORY

TECHNICAL FIELD

RGB 4-17-03

The present invention is directed to memory devices, and, more particularly, to a system and method for allowing dynamic random access memory devices to be used as cache memory.

BACKGROUND OF THE INVENTION

Memory devices are used in a wide variety of applications, including computer systems. Computer systems and other electronic devices containing a microprocessor or similar device typically include system memory, which is generally implemented using dynamic random access memory ("DRAM"). The primary advantage of DRAM is that it uses relatively few components to store each bit of data, and is thus relatively inexpensive to provide relatively high capacity system memory. A disadvantage of DRAM, however, is that their memory cells must be periodically refreshed. While a memory cell is being refreshed, read and write accesses to other rows in the memory array are blocked. The need to refresh memory cells does not present a significant problem in most applications, but it can prevent their use in applications where immediate access to memory cells is required or highly desirable.

Also included in many computer systems and other electronic devices is a cache memory. The cache memory stores instructions and/or data (collectively referred to as "data") that is frequently accessed by the processor or similar device, and may be accessed substantially faster than data can be accessed in system memory. It is important for the processor or similar device to be able to access the cache memory as needed. If the cache memory cannot be accessed for a period, the operation of the processor or similar device must be halted during this period. Cache memory is typically implemented using static random access memory ("SRAM") because such memory need not be refreshed and is thus always accessible for a write or a read memory access. However, a significant disadvantage of SRAM is that each memory cell requires a relatively large number of components, thus making SRAM data storage relatively expensive. It would be desirable to implement cache memory using DRAM

because high capacity cache memories could then be provided at relatively little cost. However, a cache memory implemented using DRAMs would be inaccessible at certain times during a refresh of the memory cells in the DRAM. For example, during refresh of a row of memory cells, it would be impossible to read data from or write data to other rows of memory cells. As a result of these problems, DRAMs have not generally been considered acceptable for use as cache memory or for other applications requiring immediate access to memory.

Attempts have been made to use DRAM as cache memory, but these attempts have not been entirely successful in solving the refresh problem so that these prior art devices are not always available for a memory access. These prior art devices have attempted to "hide" memory refreshes by including a small SRAM to store one or more rows of DRAM data during refresh of a row being addressed. However, in practice, there are still some memory access situations in which these prior art devices may not be accessed, thus suspending the operation of a processor or similar device.

There is therefore a need for a DRAM that effectively hides memory refresh under all memory access situations so that the DRAM may provide relatively inexpensive, high capacity cache memory.

SUMMARY OF THE INVENTION

A method of caching data and a cache system that may be used in a computer system includes a DRAM having a plurality of refresh blocks and a pair of SRAMs having a capacity of at least the capacity of the refresh blocks. If a block of the DRAM to which data is attempting to be written is being refreshed, the data is instead written to one of the SRAMs. When the refresh of that block has been completed, the data is transferred from the SRAM to a block of the DRAM to which data was attempted to be written. If a block to which data is attempting to be written is being refreshed and data is being transferred from the one SRAM to a block of the DRAM, the data is instead written to the other SRAM. As a result, there is always one SRAM available into which data may be written if a refresh block to which the write was directed is being refreshed.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram of a computer system containing a cache memory in accordance with one embodiment of the invention.

Figure 2 is a block diagram of a cache system that may be used as a cache memory in the computer system of Figure 1 in accordance with one embodiment of the invention.

Figure 3, is a diagram conceptually illustrating a DRAM and SRAM arrays shown in the cache system of Figure 2.

DETAILED DESCRIPTION OF THE INVENTION

Figure 1 is a block diagram of a computer system 10 that includes a processor 12 for performing various computing functions by executing software to perform specific calculations or tasks. The processor 12 is coupled to a processor bus 14 that normally includes an address bus, a control bus, and a data bus (not separately shown). In addition, the computer system 10 includes a system memory 16, which is typically dynamic random access memory ("DRAM"). As mentioned above, using DRAM at the system memory 16 provides relatively high capacity at relatively little expense. The system memory 16 is coupled to the processor bus 14 by a system controller 20 or similar device, which is also coupled to an expansion bus 22, such as a Peripheral Component Interface ("PCI") bus. A bus 26 coupling the system controller 20 to the system memory 16 also normally includes an address bus, a control bus, and a data bus (not separately shown), although other architectures can be used. For example, the data bus of the system memory 16 may be coupled to the data bus of the processor bus 14, or the system memory 16 may be implemented by a packetized memory (not shown), which normally does not include a separate address bus and control bus.

The computer system 10 also includes one or more input devices 34, such as a keyboard or a mouse, coupled to the processor 12 through the expansion bus 22, the system controller 20, and the processor bus 14. Also typically coupled to the expansion bus 22 are one or more output devices 36, such as a printer or a video terminal. One or more data storage devices 38 are also typically coupled to the expansion bus 22 to allow the processor 12 to store data or retrieve data from internal or external storage media (not shown). Examples of typical storage devices 38 include

hard and floppy disks, tape cassettes, and compact disk read-only memories (CD-ROMs).

The processor 12 is also typically coupled to cache memory 40 through the processor bus 14. In the past, the cache memory 40 was normally implemented using static random access memory ("SRAM") because such memory is relatively fast, and does not require refreshing and may thus always be accessed. However, as explained above, using SRAM for the cache memory 40 is a relatively expensive means for providing a relatively high capacity because of the large number of components making up each SRAM storage cell compared to the number of components in each DRAM storage cell.

According to one embodiment of the invention, the cache memory 40 shown in Figure 1 is implemented using a cache system 50, an example of which is shown in Figure 2. The cache system 50 includes components normally found in a DRAM, including an address decoder 52 receiving addresses through an address bus 53, a row driver circuit 54 adapted to receive row addresses from the address decoder 52, and a column driver circuit 56 adapted to receive column addresses from the address decoder 52. The row driver circuit 54 is coupled to word lines (not shown) in a memory array 60, and the column driver circuit 56 is coupled to digit lines (not shown) in the memory array 60. As shown in Figure 2, the memory array 60 is either physically or logically divided into a plurality of banks 60a-n. Each bank 60a-n is divided into one or more refresh blocks, each containing a plurality of rows that are contemporaneously refreshed. The column driver 56 is also coupled to a sense amplifier/write driver circuit 64 to route write data and read data from and to, respectively, a data input/output buffer 66 through an internal data bus 68. The data input/output buffer 66 is, in turn, coupled to an external data bus 70. As in conventional DRAMs, the cache system 50 also includes a control circuit 72 that includes a command buffer 74 receiving command signals through a command bus 76 and generating appropriate control signals for controlling the operation of the cache system 50. The control circuit 72 also includes a refresh controller 78 for refreshing the DRAM array 60 one refresh block at a time.

Unlike conventional DRAMs, the cache system 50 also includes two SRAM arrays 80, 84 that are each coupled to the sense amplifier/write driver circuit 64

to access data in the DRAM array 60. The SRAM arrays 80, 84 are also coupled to the refresh controller 78. The refresh controller 78 receives addresses from the address decoder 52, and it applies addressing and control signals to the row driver 54.

The operation of the command buffer 74, refresh controller 78 and the SRAM arrays 80, 84 in relation to the other components of the cache system 50 will now be explained with reference to the diagram of Figure 3, which conceptually illustrates the DRAM array 60 and the SRAM arrays 80, 84 shown in Figure 2. As mentioned above, the DRAM array is divided into a plurality of refresh blocks. The refresh blocks may be part of the same or different banks 60a-n of DRAM memory, or physically different DRAM devices. In the embodiment shown in Figure 3, each of the refresh blocks 61a-n has a capacity of Y bits, and each of the SRAM arrays 80, 84 also has a capacity of Y bits. Each of the refresh blocks 61a-n may be individually refreshed under control of the refresh controller 78 (Figure 2). The DRAM array 60 has twice the normal number of input/output ("I/O") lines, which are configured so that two blocks can be simultaneously accessed. As a result, it is possible for data to be read from or written to one refresh block 61a-n of the DRAM array 60 at the same time data are being transferred from one of the SRAM arrays 80, 84 to another block 61a-n of the DRAM array 60.

In operation, a read from a refresh block 61a-n that is not being refreshed is read in a conventional manner. Similarly, a write to a block 61a-n that is not being refreshed is accomplished in a conventional manner. Thus, no problem is presented in either writing to or reading from a refresh block 61a-n that is not being refreshed. In either of these cases, data access to the cache system 50 does not require any wait, thus allowing the cache system 50 to be used as a cache memory in place of a typically used SRAM without any performance limitations.

The potential problem in accessing the cache system 50 is in the event of a read or a write to a refresh block 61a-n being refreshed, and, in particular, to a different row than the row in that block that is being refreshed. The cache system 50, preferably the refresh controller 78, may check each memory command prior to initiating a refresh in a block 61a-n to determine if the memory command is a read. If a read command directed to a block 61a-n that is about to be refreshed is received, then the refresh is not initiated. In this regard, it is assumed that the duration of a refresh is

shorter than the duration of a memory read operation. Each time a read is executed, the read data are written to one of the SRAMs 80, 84. As a result, the read data are subsequently accessible in one of the SRAMs 80, 84, thereby allowing the portion of the block 61a-n that stored such data to be refreshed despite subsequent reads from that that portion. In the case of sequential reads from the rows of a block 61a-n, the reads will refresh the rows.

In the event a memory access is a write to a block 61a-n being refreshed, the write data is instead written to one of the SRAM arrays 80, 84. When the refresh of the block to which the write was directed has been completed, the refresh controller 78 starts a refresh of another block 61a-n of the DRAM array 60. While this subsequent refresh is occurring, the data that had been written to one of the SRAM arrays 80, 84 is transferred to the block 61a-n to which the earlier write was directed. If, during refresh of the second block 61a-n, a read or a write is directed toward that block 61a-n, then that data is instead stored in the other one of the SRAM arrays 80, 84. By the time the refresh of the second block 61a-n has been completed, transfer of the data from a first one of the SRAM arrays 80, 84 to the first block 61a-n will have been completed, and that SRAM array 80, 84 will be available to store write data that is subsequently directed to any other block 61a-n that is being refreshed. Therefore, a SRAM array 80, 84 is always available to store write data that is directed to a refresh block 61a-n of the memory array 60 that is being refreshed. As a result, data may always be read from or written to the cache system 50 without the need for to wait for the completion of a refresh of any block 61a-n the cache system 50. The cache system 50 may therefore be used as a cache memory in place of an SRAM that is typically used, thereby providing high capacity caching at relatively little cost.

From the foregoing it will be appreciated that, although specific embodiments of the invention have been described herein for purposes of illustration, various modifications may be made without deviating from the spirit and scope of the invention. Accordingly, the invention is not limited except as by the appended claims.